Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1-2. (Cancelled)

3. (Currently Amended) A method of generating a logic design for use in designing an integrated circuit (IC), comprising:

generating a computer instruction;

importing the computer instruction from memory; and

embedding the computer instruction within a two-dimensional schematic representation of the logic design to produce a unified database representation of the logic design, the computer instruction being devoid of and entries to a sensitivity list;

wherein the two-dimensional schematic representation includes a set of [[Register Transfer Diagrams]] register transfer diagrams (RTD).

4. (Cancelled)

5. (Previously Presented) The method of claim 3, further comprising notifying a designer when capturing data using the computer instruction violates a set of design capture rules.

6. (Cancelled)

7. (Previously Presented) The method of claim 3, further comprising generating C++ from the unified database.

8. (Previously Presented) The method of claim 7, further comprising generating Verilog from the unified database.

9. (Cancelled)

10. (Previously Presented) The method of claim 3, further comprising generating synthesizable Verilog from the unified database.

11-12. (Cancelled)

13. (Currently Amended) An article comprising a machine-readable medium which stores executable instructions to generate a logic design for use in designing an integrated circuit (IC), the instructions causing a machine to:

generate a computer instruction;

embed the computer instruction within a two-dimensional schematic representation of the logic design to produce a unified database representation of the logic design, the computer instruction being devoid of declarations and entries to a sensitivity list;

wherein the two-dimensional schematic representation includes a set of [[Register Transfer Diagrams]] register transfer diagrams (RTD).

14. (Previously Presented) The article of claim 13, further comprising instructions causing a machine to import the computer instruction.

15. (Previously Presented) The article of claim 13, further comprising instructions causing a machine to notify a designer when capturing data violates a set of design capture rules.

16. (Cancelled)

17. (Previously Presented) The article of claim 13, further comprising instructions causing a machine to generate C++ from the unified database.

18. (Previously Presented) The article of claim 13, further comprising instructions causing a machine to generate Verilog from the unified database.

19. (Cancelled)

20. (Previously Presented) The article of claim 13, further comprising instructions causing a machine to generate synthesizable Verilog from the unified database.

21-22. (Cancelled)

23. (Currently Amended) An apparatus for generating a logic design for use in designing an integrated circuit (IC), comprising:

a memory that stores executable instructions; and

a processor that executes the instructions to:

generate a computer instruction; and

embed the computer instruction within a two-dimensional schematic representation of the logic design to produce a unified database representation of the logic design, the computer instruction being devoid of declarations and entries to a sensitivity list;

wherein the two-dimensional schematic representation includes a set of [[Register Transfer Diagrams]] register transfer diagrams (RTD).

24. (Previously Presented) The apparatus of claim 23, further comprising instructions to import the computer instruction.

25. (Previously Presented) The apparatus of claim 23, further comprising instructions to notify a designer when capturing data violates a set of design capture rules.

26. (Cancelled)

27. (Previously Presented) The apparatus of claim 23, further comprising instructions to generate C++ from the unified database.

28. (Previously Presented) The apparatus of claim 27, further comprising instructions to generate Verilog from the unified database.

29. (Cancelled)

30. (Previously Presented) The apparatus of claim 23, further comprising instructions to generate synthesizable Verilog from the unified database.

31. (Previously Presented) The method of claim 3, further comprising enabling a user to change the logic design by amending the computer instruction.

32. (Previously Presented) The article of claim 13, further comprising instructions causing a machine to enable a user to change the logic design by amending the computer instruction.

33. (Previously Presented) The apparatus of claim 23, further comprising instructions to enable a user to change the logic design by amending the computer instruction.